



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/767,365	01/22/2001	Sheng Liang	2006579-0558 (CTX-199)	2538
69665 7590 05/26/2009 CHOATE, HALL & STEWART / CITRIX SYSTEMS, INC. TWO INTERNATIONAL PLACE BOSTON, MA 02110				
EXAMINER				
TRAN, QUOC A				
ART UNIT		PAPER NUMBER		
2176				
MAIL DATE		DELIVERY MODE		
05/26/2009		PAPER		

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary

Application No.

09/767,365

Applicant(s)

LIANG ET AL.

Examiner

Quoc A. Tran

Art Unit

2176

Period for Reply -- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 05 February 2009.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1, 4-6, 8-13 and 15 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1, 4-6, 8-13 and 15 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☒ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 05 February 2009 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☒ Information Disclosure Statement(s) (PTO-893)
- 4) ☐ Interview Summary (PTO-413)
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: _____
- Paper No(s)/Mail Date 02/05/2009

DETAILED ACTION

This is Final Office Action in response to the amendments/remarks filed 02/05/2009. Claims 1, 4-6, 8-13 and 15 are pending. Claims 1, 6, 8-11, 13 and 15 are independent claims. Claims 2-3, 7 and 14 are currently cancelled. The current application claims priority from provisional application No. 60/228,904 dated **08/29/2000** (by Teros, Inc).

It is noted that the objection to the specification and the 112 second paragraph set forth in the previous office action dated 11/05/2008 are hereby withdrawn.

Information Disclosure Statement

A signed and dated copy of applicant's IDS, which was filed on 02/05/2009, is attached to this Office Action.

Specification

The specification is objected to as failing to provide proper antecedent basis for the claimed subject matter. See 37 CFR 1.75(d)(1) and MPEP § 608.01(o). Correction of the following is required: the recited "*intermediary*" of Claim 15 (currently amended). The Specification does not mention the recited "*intermediary*." Thus, there is no support or antecedent basis for the recited "*intermediary*" that allows the meaning of the term to be ascertained, as required in 37 CFR 1.75(d)(1).

Claims Rejection – 35 U.S.C. 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 1, 4-6, 8-13 and 15 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Krishnamurthy et al.**, (US 20060242145A1- Division of 10/177,783- filed 04/04/2002) [hereinafter "Krishnamurthy"], in view of **Marullo et al.**, (US006044398A - filed 11/21/1997) [hereinafter "Marullo"].

Regarding **independent claim 1**, Krishnamurthy teaches:

a computer-implemented method for efficiently parsing received data file, comprising: receiving a data file; retrieving a stored version of the data file and a syntax tree comprising nodes and tokens representing data within the data file, the tree include at least one static node.

(See figure(s) 4-5 and at page 5 Paragraph [0081] → Krishnamurthy teaches this limitation, as clearly indicated in the cited text [e.g., improving TreeDiff performances by: reading and parsing a web page (e.g. a data file) into an abstract syntax tree (AST). The syntax tree is then linearized into a sequence of tokens, which consist of markup elements (defined by the markup language syntax and denoting structure, semantics, formatting or other information) and text strings that represent the content. Also

Krisnamurthy further discloses the syntax tree includes the distinguished node(s) from the old page (e.g., static node) and newpage, wherein the old page's is preserved in some form inside the new page, and if so, the subtree is, see Krisnamurthy at para [0129] and illustrates in figure 7).

comparing by comparison engine, the stored version of the data file with the received data file to identify non-matching content in the received data file,

(See Figure(s). 5 and 7 and page 5 Para [0077] --> Page 6 Para [0083]→ Krisnamurthy discloses this limitation that is defines a different between HTML pages by comparison utilized HTMLDiff algorithm and compare the parse tree or abstract syntax tree representations of the documents, using subtree equality (or some weaker measure) as a basis for comparison. In this case, a subtree representing nodes and tokens to one of the subtrees and identify only differences of the HTML to output a subtree (figure 7).)

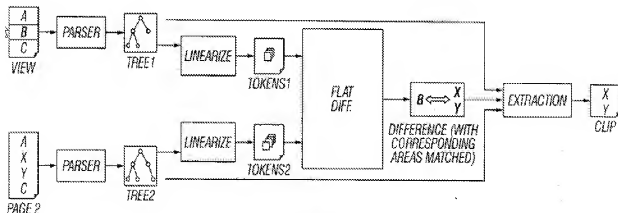


FIG. 5

parsing by a parsing engine, only the non-matching content of the received data file to form at least one subtree comprising nodes and tokens representing the non-matching content of the received data file;

(See Figure. 7 and page 8 Para [0107-0108] --> Krisnamurthy discloses this limitation, as clearly indicated in the cited text and illustrated in figure 7 [e.g., parsing only the TREEDIFF to output a sub tree of the non-matching content between the VIEW (A,B,C) and PAGE2 (A,X,Y,C), as shows in Fig. 7.)

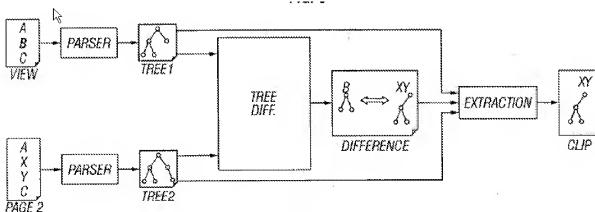


FIG. 7

replacing, at least one static node of the syntax tree with a token; and creating, a mapping from each token to one of the subtrees.

(See Figure. 5 and page 5 Para [0077] --> Page 6 Para [0083], and Para [0129] --> Krisnamurthy discloses the Improving TreeDiff Performance: Subtree matching wherein the difference algorithms compute the mapping from the nodes in T1 to the nodes in

T2. Given such a mapping, we can identify whether the distinguished node(s) from the old page is preserved in some form inside the new page, and if so, the subtree is formed)

This interpretation is supported by the applicant's disclosure, which is stated, "System 500 ... newly retrieved version of the page and the cached copy of the page are compared using a binary "diff" algorithm, which identifies the differences between the binary representation of two documents"(See current disclosure page 11 lines 11→Line 25).

In addition, Krishnamurthy does not expressly teach, but Marullo teaches:

A virtual browser,

(See column 4 lines 49-55→ Marullo discloses this limitation, as clearly indicated in the cited text [e.g., e.g., an Internet website virtual browser application automatically exercises and verifies web server applications and scripts by simulating a web browser to request, capture, store, and verify data returned from web servers, discarding data not critical to testing, and saving and reusing retained data for subsequent transactions.)

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified Krishnamurthy's method for efficiently parsing received data file that is related to the HTML *Diff* algorithm, and utilizing a proxy and some graphical user interface (GUI) code and sending the augmented page (named

page 1') to the user (named view client) (see Krisnamurthy at page 4 para[0063], to include an Internet website virtual browser as taught by Marullo, because they are from the same field of endeavor of Viewing and parsing HTML, and provides a predictable result of said automatically generating, exercising and verifying web server applications and scripts by simulating a web browser to request, capture, store, and verify data returned from web servers, discarding data not critical to testing, and saving and reusing retained data for subsequent transactions utilized a virtual browser, see Marullo at column 4 lines 49-55.

Claim 4, Krisnamurthy teaches:

wherein the data file is a web page,

(See Paragraph [006]→ Krisnamurthy discloses web page is data file.)

Claim 5, Krisnamurthy teaches:

wherein the data file is an HTML file,

(See Paragraph [006 and 0052]→ Krisnamurthy discloses data file is HTML file.)

Regarding independent claim 6,

Claim 6 is fully incorporated similar subject of claims 1 and 4-5 cited above, and are similarly rejected along the same rationale. Thus, Krishnamurthy and Marullo disclose every limitation of Claim 6 and provide proper reasons to combine, as indicated in the above rejections for Claims 1 and 4-5.

Regarding independent claim 8,

Claim 8 are fully incorporated similar subject of claims 1, and 4-5 cited above, and are similarly rejected along the same rationale. Thus, Krishnamurthy and Marullo disclose every limitation of Claim 8 and provide proper reasons to combine, as indicated in the above rejections for Claims 1, 4-5.

In addition, Krishnamurthy teaches:

a first HTML page; responsive to a determination that a cached version of the HTML page exists:

(See figure(s) 4-5 and at page 5 Paragraph [0081]→Krishnamurthy teaches this limitation, as clearly indicated in the cited text [e.g., reads and parses a web page (HTML page) from a cache into an abstract syntax tree (AST).)

responsive to a determination that the cached version of the HTML page exist: parsing, by the parsing engine, the received HTML page to form a second syntax tree comprising nodes and tokens representing the non-matching content of the received data file, the second tree containing at least one static node; and storing the second tree and the received HTML page in the cache.

(Krishnamurthy teaches retrieving a stored version of a web page and a syntax tree comprising nodes and tokens representing the data file (See figure(s) 4-5 and at page 5 Paragraph [0081] and [0129], e.g., to improving TreeDiff performances by: reading and

parsing a web page (e.g. a data file) into an abstract syntax tree (AST)). After the web page is parsed into the form of an abstract syntax tree (AST); retrieving the stored version of the data file with the received data file in the form of AST is being compared to identify non-matching content (new/dynamic content) using the TREEDIFF algorithm, (see Figure. 5 and page 5 Para [0077] --> Page 6 Para [0083], e.g., TreeDiff algorithm). In addition, Krishnamurthy is parsing only the TREEDIFF (the non-matching content between the VIEW (A, B, C) and PAGE2 (A, X, Y, C)) to form a subtree of the non-matching content. This is generally set forth at page 8 Para [0107-0108] and illustrated in figure 7 of Krishnamurthy. Also see Krishnamurthy at para [0161], discloses the recent pages and their recently extracted clips are stored in or near the machine that performs extraction enables them to be reused like a cache.)

Regarding independent claim 10,

Claim 10 is fully incorporated similar subject of claim 1 cited above, and is similarly rejected along the same rationale. Thus, Krishnamurthy and Marullo disclose every limitation of Claim 10 and provide proper reasons to combine, as indicated in the above rejections for Claim 1.

In addition, Krishnamurthy teaches:

**method of providing: identifying static content that has been
previously retrieved ...and stored in a cache, and corresponding
syntax trees that have also been stored in the cache; identifying
dynamic content that differs from the previously retrieved content;**

(See figure(s) 4-5 and at page 5 Paragraph [0081]→Krishnamurthy teaches this limitation, as clearly indicated in the cited text [e.g., reads and parses a web page (data file) from a cache into an abstract syntax tree (AST). The syntax tree is then linearized into a sequence of tokens, which consist of markup elements (defined by the markup language syntax and denoting structure, semantics, formatting or other information) and text strings that represent the content. Also Krishnamurthy further discloses the syntax tree includes the distinguished node(s) from the old page (e.g., static node) and newpage (e.g. dynamic node), wherein the old page's is preserved in some form inside the new page, and if so, the subtree is formed, see Krishnamurthy at para [0129]. Also Krishnamurthy further describes the static and dynamic web clip; see Krishnamurthy at para [0012-0013]. Also see Krishnamurthy at para [0161], discloses the recent pages and their recently extracted clips are stored in or near the machine that performs extraction enables them to be reused like a cache.)

In addition, Krishnamurthy does not expressly teach, but Marullo teaches:

derivative services from a plurality of primary service providers, comprising: a virtual browser and receiving, by a virtual browser, a request for derivative services content from a client node of a customer; retrieving, by the virtual browser, data from a plurality of primary service providers on behalf of the customer;

(See column 4 lines 49-55→ Marullo discloses this limitation, as clearly indicated in the cited text [e.g., an Internet website virtual browser (e.g., derivative services) application

automatically exercises and verifies web server applications and scripts by simulating a web browser to request, capture, store, and verify data returned from web servers (e.g., a plurality of service providers).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified Krisnamurthy's method for efficiently parsing received data file that is related to the HTML *Diff* algorithm, and utilizing a proxy and some graphical user interface (GUI) code and sending the augmented page (named page 1') to the user (named view client) (see Krisnamurthy at page 4 para[0063], to include a means of said derivative services from a plurality of primary service providers, comprising: a virtual browser and receiving, by a virtual browser, a request for derivative services content from a client node of a customer; retrieving, by the virtual browser, data from a plurality of primary service providers on behalf of the customer as taught by Marullo; because they are from the same field of endeavor of Viewing and parsing HTML, and provides a predictable result of said automatically generating, exercising and verifying web server applications and scripts by simulating a web browser to request, capture, store, and verify data returned from web servers, discarding data not critical to testing, and saving and reusing retained data for subsequent transactions utilized a virtual browser, see Marullo at column 4 lines 49-55.

Regarding independent claim 9,

Claim 9 is fully incorporated similar subject of claims 1, 4-5 and 10 cited above, and is similarly rejected along the same rationale. Thus, Krishnamurthy and Marullo disclose every limitation of Claim 9 and provide proper reasons to combine, as indicated in the above rejections for Claims 1, 4-5 and 10.

Regarding independent claim 11,

Claim 11 is fully incorporated similar subject of claim 1 cited above, and is similarly rejected along the same rationale. Thus, Krishnamurthy and Marullo disclose every limitation of Claim 11 and provide proper reasons to combine, as indicated in the above rejections for Claim 1.

In addition, Krishnamurthy teaches:

retrieving a second data file, the second data file associated with the first data file,

(See Paragraph [0033]→ Krishnamurthy discloses this limitation, as clearly indicated in the cited text [e.g., the tree traversal is performed on the second tree guided by the selected data of the first tree (e.g., second data file associated with first data file.)

Claim 12,

Krishnamurthy and Marullo teach the method of claim 11 and further comprise:

responsive to identifying non-matching content present only in the first file: adding, at least one new token to the syntax tree

(Krishnamurthy teaches retrieving the stored version of the data file with the received data file in the form of AST is being compared to identify non-matching content (new/dynamic content) using the TREEDIFF algorithm, (see Figure. 5 and page 5 Para [0077] --> Page 6 Para [0083], e.g., TreeDiff algorithm). In addition, Krishnamurthy is parsing only the TREEDIFF (the non-matching content between the VIEW (A, B, C) and PAGE2 (A, X, Y, C)) to form a subtree of the non-matching content. This is generally set forth at page 8 Para [0107-0108] and illustrated in figure 7.)

In addition, Krishnamurthy does not expressly teach, but Marullo teaches:

A virtual browser,

(See column 4 lines 49-55→ Marullo discloses this limitation, as clearly indicated in the cited text [e.g., e.g., an Internet website virtual browser application automatically exercises and verifies web server applications and scripts by simulating a web browser to request, capture, store, and verify data returned from web servers, discarding data not critical to testing, and saving and reusing retained data for subsequent transactions.)

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified Krishnamurthy's method for efficiently parsing received data file that is related to the HTML *Diff* algorithm, to include an Internet website virtual browser as taught by Marullo, because they are from the same field of endeavor of Viewing and parsing HTML, and provides a predictable result of said

automatically generating, exercising and verifying web server applications and scripts by simulating a web browser to request, capture, store, and verify data returned from web servers, discarding data not critical to testing, and saving and reusing retained data for subsequent transactions utilized a virtual browser, see Marullo at column 4 lines 49-55.

Regarding independent claim 13,

Claim 13 recites a system to implement a method recited in Claim 1.

Thus, Krishnamurthy and Marullo disclose every limitation of Claim 15 and provide proper reasons to combine, as indicated in the above rejections for Claim

1 - Also See Krishnamurthy at Para [0011], disclose databases (cache) - Also See Krishnamurthy at Para [0063] discloses a proxy that augments a web page with some graphical user interface (GUI) code and sends the augmented page (named page 1') to the user (named view client); also at Para [0036]

Krishnamurthy discloses a PAGEDIFF algorithm and at para [0109], described the use of a parser to form the Abstract Syntax Tree (AST.)

In addition, Krishnamurthy do not expressly teach, but Marullo teaches:

a virtual browser for retrieving content from content servers;

(See column 4 lines 49-55→ Marullo discloses this limitation, as clearly indicated in the cited text [e.g., an Internet website virtual browser automatically exercising and verifying

web server applications and scripts by simulating a web browser to request, capture, store, and verify data returned from web servers (e.g., a plurality of service providers).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified Krisnamurthy's method for efficiently parsing received data file that is related to the HTML *Diff* algorithm, and utilizing a proxy and some graphical user interface (GUI) code and sending the augmented page (named page 1') to the user (named view client) (see Krisnamurthy at page 4 para[0063], to include a virtual browser for retrieving content from content servers as taught by Marullo; because they are from the same field of endeavor of Viewing and parsing HTML, and provides a predictable result of said automatically generating, exercising and verifying web server applications and scripts by simulating a web browser to request, capture, store, and verify data returned from web servers, discarding data not critical to testing, and saving and reusing retained data for subsequent transactions utilized a virtual browser, see Marullo at column 4 lines 49-55.

Regarding independent claim 15,

Claim 15 recites an intermediary to implement a method recited in Claim 1. Thus, Krishnamurthy and Marullo disclose every limitation of Claim 15 and provide proper reasons to combine, as indicated in the above rejections for Claim 1 - Also See Marullo at column 4 lines 49-55, discloses a virtual browser (e.g. an intermediary).

It is noted that any citations to specific, pages, columns, lines, or figures in the prior art references and any interpretation of the references should not be considered to be limiting in any way. A reference is relevant for all it contains and may be relied upon for all that it would have reasonably suggested to one having ordinary skill in the art. See, MPEP 2123.

Response to Arguments

Applicant's arguments with respect to claims 1, 4-6, 8-13 and 13-15 have been considered but are moot in view of the new ground(s) of rejection.

It is noted; Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action (see above for details).

Further, the Examiner maintains Krishnamurthy reference. Because, Krishnamurthy teaches retrieving a stored version of the data file and a syntax tree comprising nodes and tokens representing the data file (See figure(s) 4-5 and at page 5 Paragraph [0081] and [0129], e.g., to improving TreeDiff performances by: reading and parsing a web page (e.g. a data file) into an abstract syntax tree (AST)). After the web page is parsed into the form of an abstract syntax tree (AST); retrieving the stored version of the data file with the received data file in the form of AST is being compared to identify non-matching content (new/dynamic content) using the TREEDIFF algorithm, (see Figure. 5 and page 5 Para [0077] --> Page 6 Para [0083], e.g., TreeDiff algorithm).

In addition, Krishnamurthy is parsing only the TREEDIFF (the non-matching content between the VIEW (A, B, C) and PAGE2 (A, X, Y, C)) to form a subtree of the non-matching content. This is generally set forth at page 8 Para [0107-0108] and illustrated in figure 7 of Krishnamurthy.

"What matters is the objective reach of the claim. If the claim extends to what is obvious, it is invalid under § 103." KSR Int'l Co. v. Teleflex, Inc., 127 S. Ct. 1727, 1742 (2007). To be nonobvious, an improvement must be "more than the predictable use of prior art elements according to their established functions." Id. at 1740.

In this case, the Examiner's analysis of Krishnamurthy's teachings relate to a **PROXY**, that augments a web page with some graphical user interface (GUI) code and sends the augmented page (named page 1') to the user (named view client) (Krishnamurthy at Para [0063]).

As recognized by the Examiner, Krishnamurthy does not teach the use of **virtual browser**. On the other hand, in what is fairly characterized as analogous art in accordance with the above-noted case law, **Marullo** (new reference) discloses an Internet website virtual browser automatically exercises and verifies web servers contents (applications and scripts) by simulating a web browser to request, capture, store, and verify data returned from web servers, discarding data not critical to testing, and saving and reusing retained data for subsequent transactions in a virtual dynamic web browser environment (e.g. a proxy environment). This is generally set forth at

column 4 lines 49-55 of Marullo.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified Krisnamurthy's method for efficiently parsing received data file that is related to the HTML *Diff* algorithm, and utilizing a proxy and some graphical user interface (GUI) code and sending the augmented page (named page 1') to the user (named view client) (see Krisnamurthy at page 4 para[0063], to include an Internet website virtual browser as taught by Marullo, because they are from the same field of endeavor of Viewing and parsing HTML, and provides a predictable result of said automatically generating, exercising and verifying web server applications and scripts by simulating a web browser to request, capture, store, and verify data returned from web servers, discarding data not critical to testing, and saving and reusing retained data for subsequent transactions in a virtual dynamic web browser environment (e.g. a proxy environment), see Marullo at column 4 lines 49-55.

Thus, Krisnamurthy and Marullo clearly disclose identifying non-matching or dynamic content and parsing only the non-matching or dynamic content to form at least one subtree as claimed (see the Applicant's remarks at page 9 lines 5-7) and provided proper reasons to combine.

Conclusion

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP

§ 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Quoc A. Tran whose telephone number is 571-272-8664. The examiner can normally be reached on Mon through Fri 8AM - 5PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Doug Hutton can be reached on (571)272-4137. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/Quoc A, Tran/
Patent Examiner

/DOUG HUTTON/
Supervisory Patent Examiner, Art Unit 2176